

REMARKS

Please reconsider the application in view of the above amendments and the following remarks. Applicants thank the Examiner for carefully considering this application.

Disposition of Claims

Claims 40, 41, and 43-56 were pending in this application. Claims 40, 43, 47, and 52 are independent. The remaining claims depend, either directly or indirectly, from claims 40, 43, 47, and 52.

Claim Amendments

Claims 40, 41, 43, 46, 47, 48, 52, and 53 are amended for clarity. No new matter is added by way of the amendments as support for these amendments may be found, for example, in paragraphs [1026]-[1048] of the Specification. Further, claim 56 is amended to remove an extraneous period.

Drawings

Applicants respectfully request that the Examiner indicate that the drawings filed April 11, 2000 are accepted in the next action.

Information Disclosure Statement (IDS)

The Examiner indicated that an IDS filed on May 29, 2007 was not considered by the Examiner because the IDS failed to comply with the provisions of 37 C.F.R. §§ 1.97, 1.98 and MPEP § 609. Specifically, the Examiner stated that two of the references were previously cited by the Examiner and that copies of the cited non-patent literature were not found in the application file. As discussed with the Examiner on October 30, 2007, the form PTO-1449 provided with the Applicants' response filed on May 29, 2007 was not a new IDS, but rather a copy of the form PTO-1449 from the IDS filed on April 13, 2004 which has not been considered by the Examiner in any of the subsequent office actions. Furthermore, the unconsidered IDS was filed before the office action in which the Examiner cited the two references at issue, *i.e.*, the Office Action dated April 21, 2004, and copies of the cited non-patent literature are available in PAIR. Accordingly, Applicants respectfully request that the Examiner consider and initial each reference listed on the form PTO-1449 filed on *April 13, 2004*.

Claim Objections

The Examiner objected to claim 56 because of an informality, *i.e.*, that an extra period was included at the end of the claim. Claim 56 is amended herein to remove the extra period. Accordingly, withdrawal of this objection is respectfully requested.

Rejections under 35 U.S.C. § 103

Claims 40, 41, and 43-56 are rejected under 35 U.S.C. §103(a) as being unpatentable over Janice M. Stone's "A Simple and Correct Shared-Queue Algorithm Using Compare-and-Swap," ©

1990 (hereinafter “Stone”) in view of Kruse, Leung, and Tondo’s Data Structures and Program Design in C, © 1991 (hereinafter “Kruse”). To the extent that this rejection may still apply to the amended claims, this rejection is respectfully traversed.

To establish a *prima facie* case of obviousness, “[f]irst, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) *must teach or suggest all the claim limitations*.” MPEP § 2143 (emphasis added). Further, “all words in a claim must be considered in judging the patentability of that claim against the prior art.” MPEP § 2143.03. Applicants assert that the references when combined fail to teach or suggest all of the limitations of the amended claims.

One or more embodiments of the invention are directed to a concurrent double-ended deque implemented as a circular buffer on a bounded-size contiguous array. Elements may be concurrently added to and/or removed from both ends of the deque. See Specification at paragraphs [1026]-[1031]. For convenience of description, one end is designated as the left end and the other end is designated as the right end. Indices into the array, referred to as L and R, refer to the next location in the array that may be considered for use when adding an element to the deque at the left or right end of the deque, respectively. *Id.* at paragraph [1034]. In other words, these indices do not point to the current last element in the deque at their respective ends of the deque but rather to an element of the array that is adjacent to the current last element at their respective ends. Further, the array element to the left of the element indicated by R or the right of the element indicated by L always contains a non-null value of the deque unless the deque is empty. *Id.* at paragraph [1034].

In addition, if the array element indicated by R or by L contains a non-null value, then the deque is full. *Id.* at paragraph [1034]. This approach to managing left and right indices allows access operations, *i.e.*, pushes and pops, at each end of the deque to determine whether the deque is empty or full without regard to the relative locations of the indices. *Id.* at paragraph [1048].

Turning to the rejection of the claims, amended claim 40 requires a concurrent double-ended deque implemented as a circular buffer on a contiguous array of bounded size delimited by a first end identifying index and a second end identifying index. Further, the claim specifically states that elements may be added to and removed from both ends of the deque. In addition, the claim requires that the first end identifying index identifies an element of the *array* that is adjacent to the first end element of the *deque* and that the second end identifying index identifies an element of the *array* that is adjacent to the second end element of the *deque*. In other words, each end-identifying index refers to an element in the array that is adjacent to the element at the end of the deque, not to the element at the end of the queue. The claim also requires a function sequence that includes a DCAS operation operable to update an element of the array and one of the end identifying indices.

1. The cited references do not teach or suggest a concurrent double-ended deque.

Stone is directed to an implementation of a concurrent shared first-in-first-out queue that is “structured as a one-way linked list, with insertion at the end and deletion from the beginning of the list.” Stone at page 495, Introduction, paragraph 3. Kruse is directed to implementations of a non-current, non-shared queue implemented on a circular array with two indices, one index that tracks the front of the queue and one index that tracks the rear of the index. In the implementations presented in Kruse, insertion occurs at the end or rear of the queue and deletion occurs at the beginning or front of the queue. See Kruse at pages 73-75. Both Stone and Kruse are completely

silent regarding any type of queue in which insertions and deletions may occur at both ends of the queue. Accordingly, neither Stone nor Kruse, nor the combination of the two, can possibly be read to teach or suggest a concurrent double-ended deque in which elements may be added and removed at both ends of the deque as required by claim 40.

2. The cited references do not teach or suggest a first end identifying index that identifies an element of the array adjacent to a first end element of the deque and a second end identifying index that identifies an element of the array adjacent to a second end element of the deque.

The Examiner admits that Stone does not teach end identifying indices that identify an element adjacent to one of two end elements of the deque and instead relies on Kruse to teach these limitations. See Office Action dated August 23, 2007 at page 4. As is clearly shown in the C functions disclosed by Kruse that implement a queue, the front index of the queue is the index of the element in the underlying array that contains the front element of the queue, and the rear index of the queue is the index of the element of the underlying array that contains the rear element of the queue. See Kruse at pages 73-75. In other words, the front and rear indices of the queue identify the end elements of the queue, not elements of the array adjacent to the end elements of the queue. Accordingly, neither Stone nor Kruse, nor the combination of the two, can possibly be read to teach or suggest the first and second end identifying indices as recited in claim 40. The only way the Examiner can make such an assertion is by mischaracterizing the prior art.

3. The cited references do not teach or suggest a DCAS operation operable to update an element of the array and one of the end identifying indices.

Stone discloses using a Compare-and-Swap-Double (CSDL) instruction in the algorithms in the algorithms to add (*i.e.*, enqueue) an element to the tail of a concurrent shared FIFO queue and remove (*i.e.*, dequeue) an element from the head of the concurrent shared FIFO queue. Specifically,

in the enqueue algorithm, the CSDBL instruction is used to update the tail pointer of the FIFO queue and a counter associated with the tail pointer. See Stone at page 498-499 and Figure 2. And, in the dequeue algorithm, the CSDBL instruction is used to update the head pointer of the FIFO queue and a counter associated with the head pointer. See Stone at page 499 and Figure 3. Stone is completely silent regarding using the CSDBL instruction to update (i) an element of the array or (ii) both an element of the array and either the head pointer or the tail pointer. In addition, Kruse is completely silent regarding to use of DCAS operations. Accordingly, neither Stone nor Kruse, nor the combination of the two, can possibly be read to teach or suggest a DCAS operation operable to update an element of the array and one of the end identifying indices as required by claim 40. The only way the Examiner can make such an assertion is by mischaracterizing the prior art.

4. Claims 40, 41, and 43-56 are patentable over the cited references.

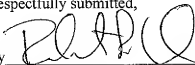
In view of the above, Stone and Kruse, whether considered separately or in combination, fail to render amended independent claim 40 obvious. Amended independent claims 43, 47, and 52 include limitations substantially similar to one or more of those discussed above for claim 40. Thus, claims 43, 47, and 52 are patentable over Stone and Kruse for at least some of the same reasons. Claims 41, 44-46, 48-51, and 53-56 depend directly or indirectly from claims 40, 43, 47, and 52 and are thus allowable over Stone and Kruse for at least the same reasons. Accordingly, withdrawal of this rejection is respectfully requested.

Conclusion

Applicants believe this reply is fully responsive to all outstanding issues and places this application in condition for allowance. If this belief is incorrect, or other issues arise, the Examiner is encouraged to contact the undersigned or his associates at the telephone number listed below. Please apply any charges not covered, or any credits, to Deposit Account 50-0591 (Reference Number 03226/959002).

Dated: November 21, 2007

Respectfully submitted,

By 

Robert P. Lord
Registration No.: 46,479
OSHA · LIANG LLP
1221 McKinney St., Suite 2800
Houston, Texas 77010
(713) 228-8600
(713) 228-8778 (Fax)
Attorney for Applicants